INDIAN INSTITUTE OF TECHNOLOGY MADRAS

Department of Chemical Engineering

CH5350 : Applied Time Series Analysis (Jul-Nov 2018)
Solutions to Assignment 5

Marks Distribution

	1	2	3	4	5
Marks	25	20	25	20	10

1 Model selection criteria - AIC

The R script is given below:

```
ar_order = {}
ma_order = {}
ar_arma_order = {}
ma_arma_order = {}
ar_aic = {}
ma_aic = {}
arma_aic = {}

N = 6000
for(i in 1:100){
# Simulate 100 realizations for the given process
wk = arima.sim(model = list(ar = c(0.2,0,0.1)),n = N)

# Fit best AR, MA and ARMA models for each of the realizations
mod_ar = auto.arima(wk,max.p = 10, max.q = 0,ic = "aic");
mod_ma = auto.arima(wk,max.p = 0, max.q = 10,ic = "aic");
mod_arma = auto.arima(wk,max.p = 10, max.q = 10,start.p = 1,start.q = 1,ic = "
    aic");

# Store the best orders of AR, MA and ARMA for each of the noise realizations
```

```
ar_order[i] = length(mod_ar$model$phi)
ma_order[i] = length(mod_ma$model$theta)
ar_arma_order[i] = length(mod_arma$model$phi)
ma_arma_order[i] = length(mod_arma$model$theta)

# Store minimum AIC values for each of the model structures for all the noise
    realizations
ar_aic[i] = mod_ar$aic
ma_aic[i] = mod_ma$aic
arma_aic[i] = mod_arma$aic
}
ar_ma_arma_aic = cbind(ar_aic,ma_aic,arma_aic)
# Indexing appropriate model based on minimum AIC for each realization
min_aic_struct = as.matrix(apply(ar_ma_arma_aic, 1, which.min))
# Frequency with which each of the model structures are identified across
    realizations
table(min_aic_struct)
# Determining the realization indices where the optimal model is AR
index1 = which(min_aic_struct %in% c(1))
# frequency with which true AR order is identified
freq_AR3 = length(which(ar_order[index1] %in% c(3)))
```

For 100 set of realizations, the best model within the same model structure for each realization and also across model structures are identified based on minimum AIC value. For the case of $N = 600$, out of 100 realizations, the best model is identified as AR: 46 times, MA: 28 times and ARMA: 26 times. The numbers may vary depending on realizations. In our case, the model structure as well as the order matches with the generating process (AR(3)) only for 10 realizations across 100. From this it may be inferred that although AIC gives a good guess for the model order, it is necessary to perform other statistical checks such as residual and significance check for the estimated parameters in order to come up with the best model.

For the case of $N = 100$, out of 100 realizations, the best model is identified as AR: 54 times, MA: 22 times and ARMA: 24 times. The frequency with which AR(3) model is identified is 0. So, it can be inferred that larger the sample size, more is the ease (based on some criterion) in identifying the true process.

`Note:` For finding the best model based on AIC, `auto.arima` routine in R has been used. There is small issue wrt ARMA models. The AR and MA orders for the case of ARMA are minimally given as 1 (initializing `start.p, start.q`) so that it does n't identify AR or an MA model. But it has been observed that the routine fails some times. Although it misses

best ARMA structure sometimes, it does n't effect the results corresponding to best model across model structures and the frequency with which the correct structure and order are identified.

2.

(a).

The data is shown in Figure 1

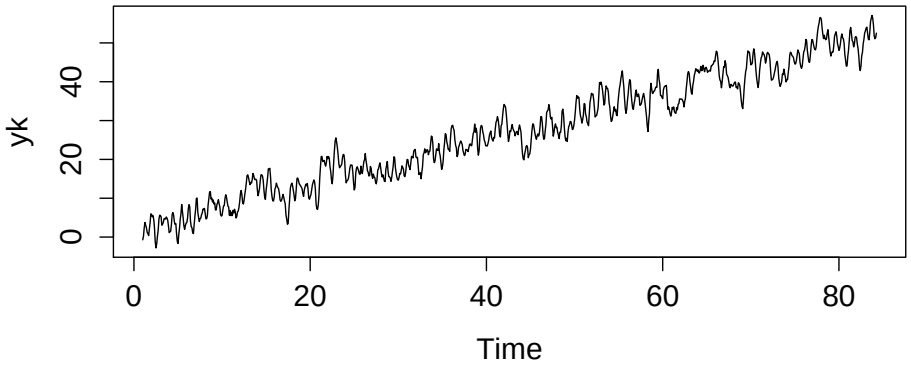


Figure 1: Cardiovascular data

Clearly, the data has some seasonal component. The data is decomposed using stl command and is shown in Figure 3

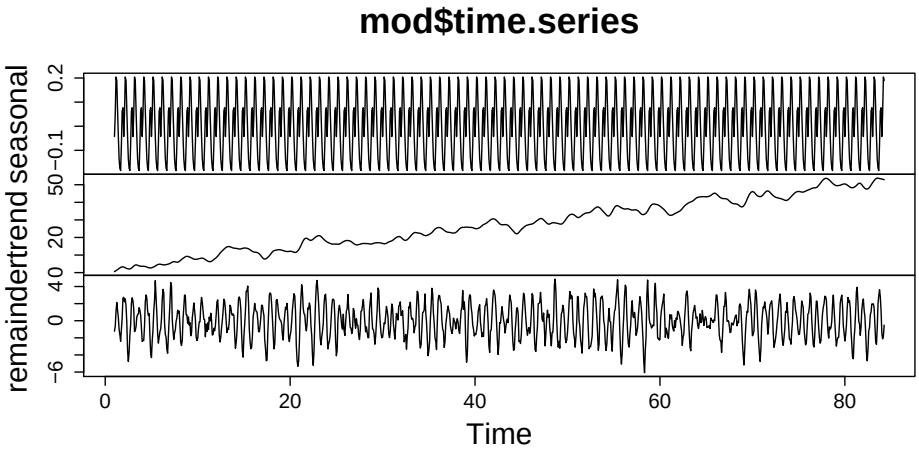


Figure 2: SARIMA data

The acf of remainder series is shown in Figure 4

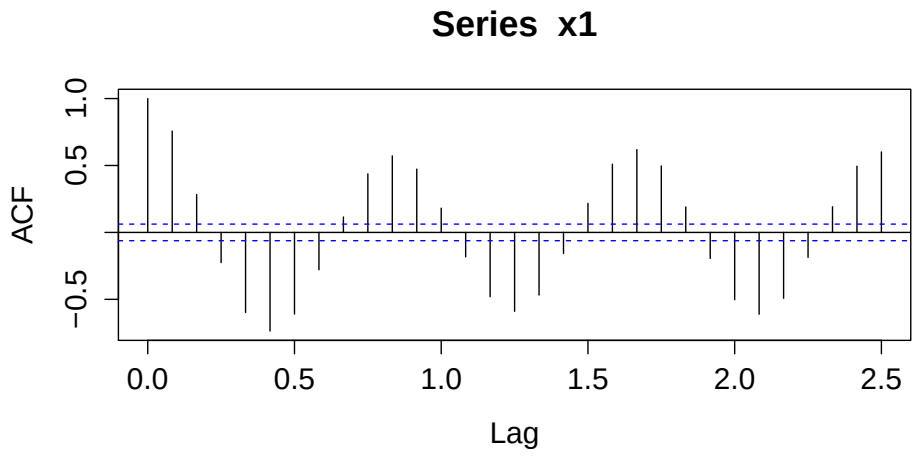


Figure 3: decomposed data

Clearly, the signal is periodic with a period of 10 samples. Hence a sum of cosine and sine signal is fitted using lm command. The acf of residuals is shown in Figure 5

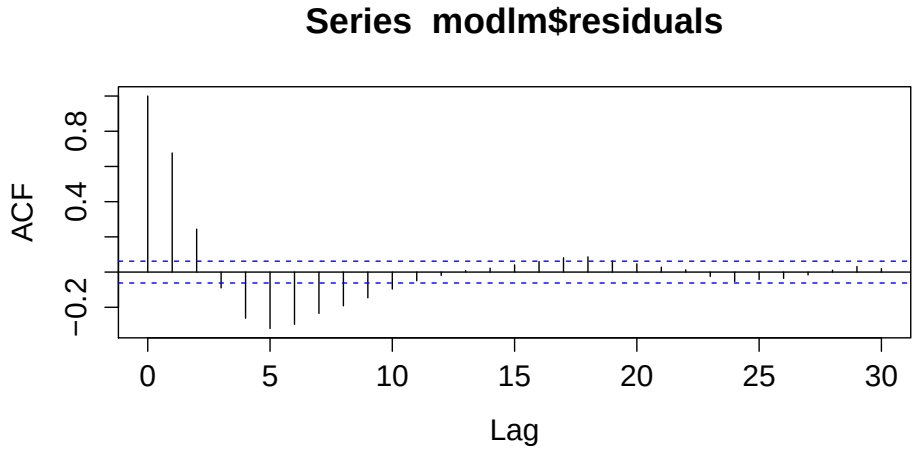


Figure 4: ACF of residuals

The pacf of residuals is shown in Figure 6

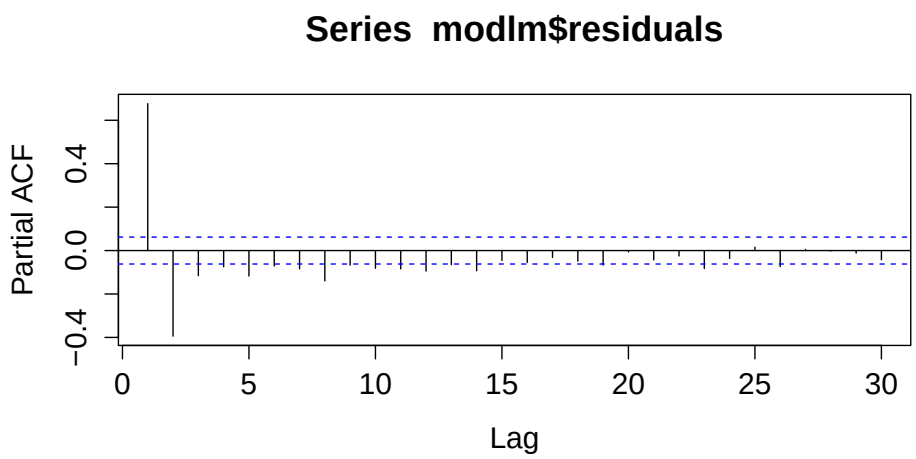


Figure 5: PACF of residuals

The pacf plot conforms that the process is of AR(2) model. Fitting an AR(2) model gives the coefficients as

arima(data1, order = c(2, 0, 0))

Coefficients:

	ar1	ar2
	-0.9434	-0.3940
s.e.	0.029	0.029

The ACF of residuals is shown in Figure 7

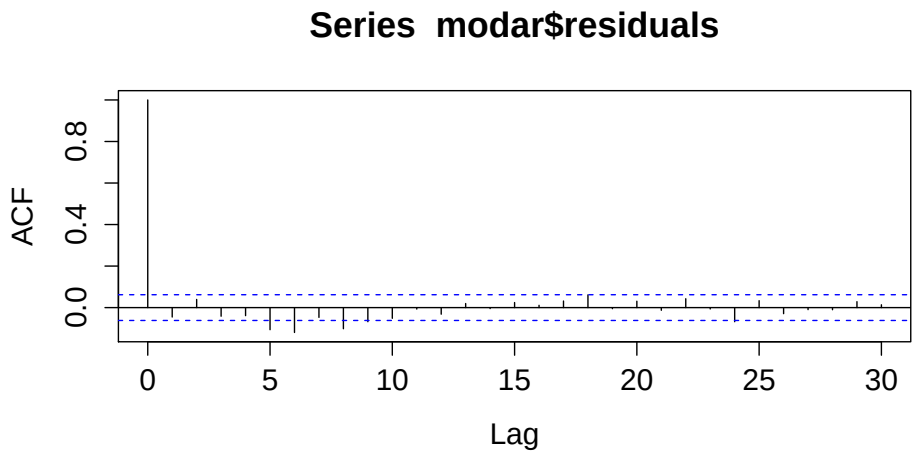


Figure 6: ACF of residuals

Clearly the ACF plot resembles like a white noise process. Hence the process is modelled as

$$x[k] = 2.22\sin(0.2\pi k) + v[k] \quad \text{where} \quad v[k] = -0.9434v[k-1] - 0.394v[k-2] + e[k]$$

(b).

The ACF of the data is shown in Figure 8

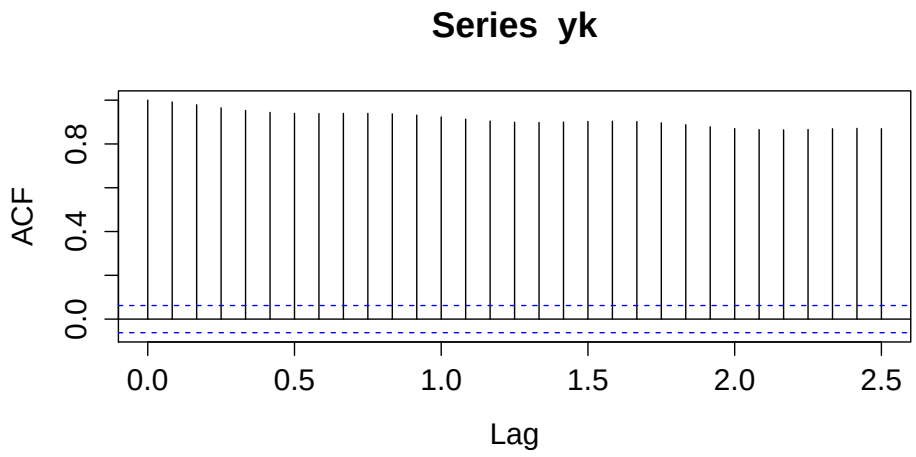


Figure 7: ACF of sarima data

Clearly, it shows non-stationarity. Hence the series is differenced and the acf of differenced series is shown in Figure 9

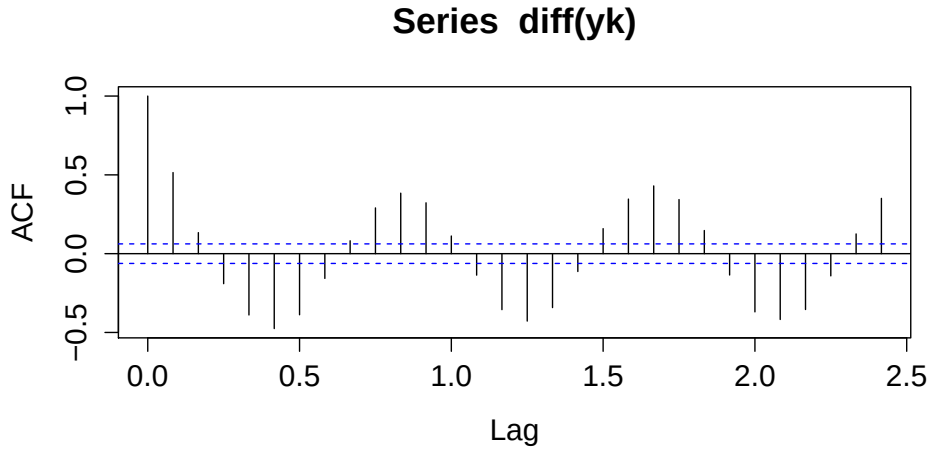


Figure 8: ACF of differenced series

It has seasonality of 10 samples. Hence the series is modelled using arima command in R as mod=arima(yk,order=c(1,0,0),seasonal=list(period=10,order=c(1,0,1)))

Coefficients:

	ar1	ar2	sar1	sma1
	1.4971	-0.5041	-0.5242	0.6545
s.e.	0.0276	0.027	0.1119	0.0982

The acf plot of residuals is shown in Figure 10

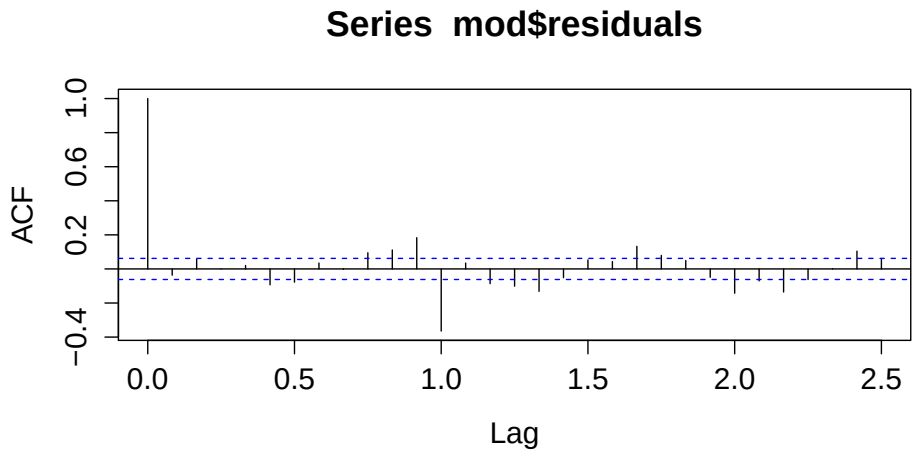


Figure 9: ACF of residuals

ACF plot almost resembles like that of white noise process except for a few coefficients which are significant at higher lags. Hence the final model is

$$\hat{x}[k] = (1.3893x[k-1] - 0.557x[k-2]) \times (0.5242x[k-10] - 0.6545e[k-10])$$

3.

Given
A RV X follows a uniform distribution over the interval [0, θ], $X \sim U[0, \theta]$
Therefore PDF is $f_{X_i}(X_i, \theta) = \frac{1}{\theta}, 0 \leq X_i \leq \theta$

(a) Likelihood function

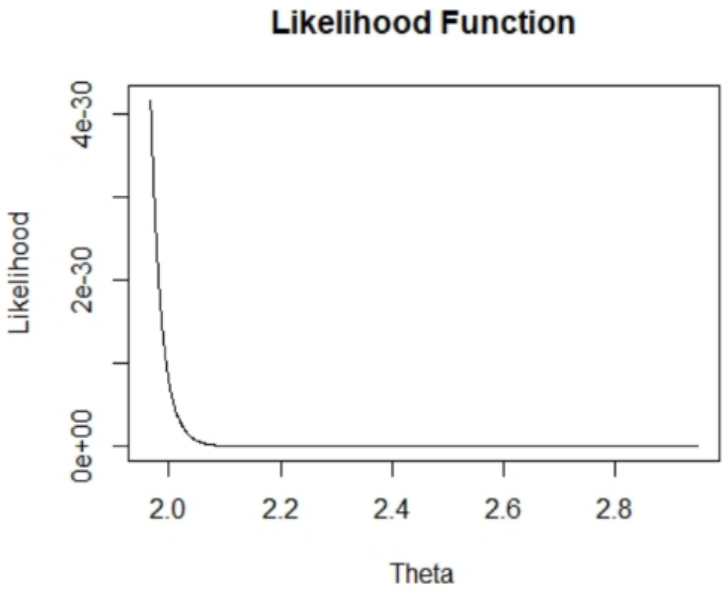
$$L(\theta; X_i) = \prod_{i=1}^{N} f_{X_i}(X_i, \theta) \text{(Since the observations are independent)}$$

$$L(\theta; X_i) = (\frac{1}{\theta})^N$$

(b) Theoretical ML estimate of θ

Maximizing likelihood implies minimum θ
But we know $min_\theta = max(X_1, X_2, ----, X_N)$

(c) Determining optimal estimate of θ for given data



Optimal Estimate of θ is 1.966975

(d) Theoretical bias in MLE Estimate

$$\begin{aligned} E[\hat{\theta}] &= E[max(X_i)] \\ &= \int_0^{\theta} (1 - (\frac{x}{\theta})^N) dx \\ &= \frac{N\theta}{N+1} \end{aligned}$$

$$\text{Therefore, Bias is } = E[\hat{\theta} - \theta] = \frac{-\theta}{N+1}$$

(e) Challenge with different distribution

If $X_i \sim U[0, \theta)$,then PDF is $f_{X_i}(X_i, \theta) = \frac{1}{\theta}, 0 \leq X_i < \theta$
Therefore, Maxima doesn't exist and it is impossible to determine ML estimate of θ

4.

Given exponential distribution

$$f(y) = \lambda e^{-\lambda y}$$

For the estimator to be efficient the function

$$\hat{\theta} = \frac{s(\theta)}{I(\theta)} + \theta$$

is to be independent of theta.

(a) $\hat{\theta} = \lambda$

The log-likelihood function is given by

$$L = \ln f(y) = \ln \lambda - \lambda y$$

Assuming the parameter to be estimated is $\theta = \lambda$, the score function is obtained as

$$S(\lambda) = \frac{\partial L}{\partial \theta} = \frac{1}{\lambda} - y$$

The information matrix is obtained as

$$\begin{aligned} I(\lambda) &= -E(\frac{\partial^2 L}{\partial \lambda^2}) \\ &= \frac{1}{\lambda^2} \end{aligned}$$

Here $\hat{\theta}$ is obtained as

$$\hat{\theta} = \lambda^2 y$$

which is a function of λ. Hence there exists no efficient estimator to estimate λ.

(b) $\hat{\theta} = \frac{1}{\lambda}$

The log-likelihood function is given by

$$L = \ln f(y) = \ln \lambda - \lambda y$$

Assuming the parameter to be estimated is $\theta = \frac{1}{\lambda}$, the score function is obtained as

$$S(\frac{1}{\lambda}) = \frac{\partial L}{\partial \theta} = -\lambda + \lambda^2 y$$

The information matrix is obtained as

$$\begin{aligned} I(\lambda) &= -E(\frac{\partial^2 L}{\partial \lambda^2}) \\ &= \lambda^2 \end{aligned}$$

Here $\hat{\theta}$ is obtained as

$$\hat{\theta} = y$$

which is a not a function of $\frac{1}{\lambda}$. Hence there exists an efficient estimator to estimate $\frac{1}{\lambda}$.

5.

Let x[k] be a stationary process. The sample mean from N samples is given by

$$\bar{X} = \frac{1}{N}\sum_{i=1}^{N} x_i[k]$$

$$\begin{aligned} \text{Therefore,Var(x)} &= E(\bar{X}\bar{X}) \\ &= \frac{1}{N^2}E(\sum_{i=1}^{N} x_i[k] \sum_{j=1}^{N} x_j[k]) \\ &= \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} E(x_i[k]x_j[k]) \\ &= \frac{1}{N^2}\sum_{\substack{i=1 \\ i=j}}^{N} E(x_i[k]x_i[k]) + \frac{1}{N^2}\sum_{\substack{i=1 \\ i\neq j}}^{N}\sum_{j=1}^{N} E(x_i[k]x_j[k]) \\ &= \frac{1}{N^2}N\sigma_{xx}[0] + \frac{2}{N^2}\sum_{i=1}^{N}\sum_{j=i+1}^{N} E(x_i[k]x_j[k]) \\ &= \frac{1}{N^2}N\sigma_{xx}[0] + \frac{2}{N^2}[(N-1)\sigma_{xx}[1] + (N-2)\sigma_{xx}[2] + - - - - - - - - - +(1)\sigma_{xx}[N-1]] \\ &= \frac{1}{N^2}N\sigma_{xx}[0] + \frac{2}{N^2}\sum_{l=1}^{N-1}(N-l)\sigma_{xx}[l] \\ &= \frac{1}{N}[\sigma_{xx}[0] + 2\sum_{l=1}^{N-1}(1-\frac{l}{N})\sigma_{xx}[l]] \end{aligned}$$

Using Monte Carlo simulations for the given MA(1) process, we get 0.003901444
By using c as 0.4 in the above derived expression, we get 0.0039168